

From: John Giannandrea
To: Microsoft ATR
Date: 1/28/02 4:52am
Subject: Microsoft Settlement

Attached in HTML and Plain text.
http://www.meer.net/jg/doj_comments.html

jg@meer.net

Comments on the Revised Proposed Final Judgment

http://www.meer.net/jg/doj_comments.html

John Giannandrea, Independent Software Developer,
Formerly ('94-'99) Chief Technologist in the Internet Browser group at Netscape/AOL

Summary

After reviewing the Revised Proposed Final Judgment, the Competitive Impact Statement, the May 18th 1998 Antitrust complaint together with the findings of the District Court and the Court of Appeals I submit that the Proposed Final Judgment fails to describe effective remedies for Microsoft's illegal activities.

An effective Final Judgment would prevent recurrence of the illegal behavior and provide relief and protection for independent software developers to develop innovative new middle-ware products and compete with Microsoft in the market for Windows software. The terms of this Final Judgment will not achieve this result because it is seriously flawed.

These comments briefly describe the following problems with the Proposed Final Judgment:

1. Problems with the scope of the remedy
2. Shortcomings in the OEM configuration provisions
3. Loopholes and technical shortcomings with the wording of the judgment
4. Restrictive language related to Intellectual Property.
5. Problems with the term and proposed implementation
6. Flaws in several of the definitions

Taken together I believe these flaws in Proposed Final Judgment make it an inappropriate remedy for the illegal behaviors found by the Court of Appeals. While changing some of the specific wording of the Final Judgment and removing some of the loopholes will make it stronger, on balance it is a wholly inappropriate remedy for the ongoing harm done by Microsoft in protecting and extending its Windows monopoly.

jg@meer.net
January 27th, 2002.

1. Problems with the scope of the remedy

There are several problems with the scope of the proposed remedies which are likely to make it ineffective in practice. The Final Judgment does not correct the harm done to the marketplace today by Microsoft's existing software products, nor address the issue of backwards compatibility and harm done to the market by ongoing changes ("upgrades"). Nor does the Final Judgment address the crucial issue of APIs in Microsoft middle-ware products themselves, as opposed to APIs in the Windows Operating System Product.

1.1 What products fall under the proposed remedy?

Sections III.D, III.E and III.H limit the practical effects of the Final Judgment to some future versions of Microsoft's latest operating system product (WindowsXP, SP1) or 12 months from submission of the Final Judgment. This will not provide effective remedy for the actual installed base of Windows users, of which WindowsXP remains a small minority. Microsoft's monopoly position is, and will be for the length of the initial proposed term, made up of Windows2000, WindowsME, Windows98 and Windows95 products and their associated middle-ware product lines. It is in these products that harm is and was being caused by the illegal activities. For the Final Judgment to be effective in providing relief, the communications protocol and Windows API disclosures need to apply to the actual installed base of Windows. It is no more technically difficult for Microsoft to document current APIs than it is to do so in future products.

The final paragraph of III.H limits the proposed remedies to middle-ware as defined by a timeline relative to the release of new Windows operating system products. The reality is that the illegal conduct relates to all existing and past Microsoft middle-ware products, and the release of future versions of Windows will not significantly affect the harm being done in the marketplace. There is no technical reason why existing Microsoft and non-Microsoft middle-ware will not be compatible with future versions of Windows. In fact Microsoft makes considerable effort to ensure that Windows is "backwards compatible" with its own applications. Remedies need to apply to all future versions of Windows, and all middle-ware now and in the future, and the obligations of the monopoly holder should not change unilaterally with a product release cycle under their express control. Much of the harm found by

WTC-00027579_0002

the Court is related not just to the disclosure of interfaces and APIs, but to the fact that Microsoft can stop supporting a documented feature or API without consulting the affected parties.

One possible way to improve the Final Judgment would be to add a new condition to III. C. that allows OEMs the option of shipping any prior Microsoft middle-ware with any subsequent version of Windows.

1.2 Middle-ware APIs are as important as Windows APIs

Section III.D. proposes that Microsoft shall disclose APIs used by its middle-ware to interoperate with a Windows operating system. Since middle-ware such as Internet Explorer or Windows Media Player has added, subtracted or altered significant APIs with each subsequent version, including minor, so called "maintenance" versions, and since these APIs are depended on by the the majority of ISVs. III.D. should be extended to require disclosure of all APIs used by, or *provided by* any Microsoft middle-ware product, including APIs in other middle-ware software.

1.3 Changes to current and past middle-ware needs to be covered

The definition in VI.J excludes software in minor version changes from the definition of Microsoft middle-ware. Yet it was exactly such a minor change that disabled Java for millions of Internet Explorer users, or forced thousands of ISVs to abandon the Web Plug-in API and redevelop or abandon their middle-ware. (See <http://www.meer.net/jg/broken-plugins.html>)

At a minimum all software middle-ware released by Microsoft and in use by a majority of Windows users should be covered by the Final Judgment for it to be effective.

2. Shortcomings in the OEM configuration provisions

It is clear from the findings of the Court that there needs to exist remedies that enable OEMs and End Users to be able to add, remove and replace middle-ware without limitation by Microsoft through its Windows product. It has been shown to the Court that its technically easy to allow middle-ware either from Microsoft or its competitors to be added and removed from the Windows operating system. The current language in the Final Judgment does not protect distribution of new and innovative forms of middle-ware and therefore fails to remedy the current situation where investment and competition in Windows middle-ware is "chilled" by Microsoft's prior and current practices.

III.H.3 allows Microsoft to undo an OEM configuration in any subsequent version of a Windows product and to change the way an OEM's configuration interacts with Windows in each subsequent version. This lack of "backwards compatibility" is in Microsoft's interest at the expense of the OEM's investment.

III.H.3. Allows Windows OS to undo an OEM's configuration automatically after 14 days. But it does not give the same capability to an ISV, or the OEM themselves. If a third party provides competitive differentiation by adding features and services on top of Windows they should be able to do so with no hindrance from Microsoft at all. If it is determined that Windows should have a "revert" feature that disables or undoes an OEM's enhancements, then that feature should have an "undo" capability so that the enhanced product purchased from the third party is not irreparably harmed by the behavior of the Windows software at some later time.

III.H attempts to give end users and OEMs the right to add and replace non Microsoft middle-ware with competitive middle-ware, an essential component of the proposed remedies. Rather than just stating this as a simple requirement, additional restrictions are imposed in III.H.2:

- that competing middle-ware be replacing a Microsoft middle-ware
- that the middle-ware be a specific subset of possible middle-ware that has a particular and limited type of user interface
- that Microsoft can require (and itself present?) a confirmation dialog for the end user if the change is made by software that the user presumably installed themselves

III.H.3 imposes conditions on Microsoft operating system products altering OEM configurations, but Microsoft middle-ware also has a documented history of making such alterations. The Final Judgment does not protect OEM investments or end user choices unless it enjoins all Microsoft software products from altering, without express permission, the end user experience. It is exactly Microsoft's ability to make unilateral changes that expresses its monopoly power and distorts the market for improvements to Windows.

The mechanism proposed in III.H.1 allows Microsoft to provide a interface choice to enable "all Microsoft Middle-ware Products as a group". This should be specifically disallowed since it reinforces the distinction between Microsoft and non Microsoft software, and suggests that an end user would be given the default choice of "taking everything" (i.e. all available Microsoft middle-ware,

turning off competitors middle-ware) in order to allow ease of use and configuration.

III.C.3 The requirement that a non-Microsoft middle-ware product should display a user interface "of similar size and shape" to a Microsoft middle-ware product is technically onerous. The additional inferred requirement that a middle-ware product can only launch automatically if a Microsoft middle-ware product were otherwise to do so, is also technically unreasonable. If the purpose of this remedy is to allow competition in such middle-ware; to allow, for example, an OEM to configure a PC so that it connected automatically to an IAP or ICP on boot up, then these restrictions would preclude this.

3. Loopholes and technical shortcomings with the wording of the judgment

There are significant exceptions and conditions attached to the definitions used by the Final Judgment. These exceptions appear to make the remedies themselves weaker and in several cases are technically inaccurate or groundless.

3.1 Excluding existing middle-ware

Section III.H after III.H.3 describes two exceptions where Microsoft middle-ware would be allowed to execute in preference to competing Middle-ware. These exceptions effectively negate the value of III.H and are seriously flawed.

3.1.1 The first exception is for middle-ware "invoked solely for use in inter-operating with a server maintained by Microsoft". Given the current and past scope of MSN and the services provided by various servers in the "microsoft.com" domain, this exception is unreasonable. For example, a component of Windows that contacted a server to upgrade or maintain the device driver software on a Personal Computer would be exempt from III.H. This would presumably preclude an OEM from providing their own value-add service using the same component APIs of Windows. As the value and prevalence of network services grows, Microsoft would be able to continue to exclude competing middle-ware as long as they could define the service as being hosted at Microsoft. This would also include most .NET services, which Microsoft has publicly stated will be at the core of most end user functions in all future versions of Windows. The proposed remedy for past behavior is ineffective.

3.1.2 The second exception is if "non-Microsoft middle-ware fails to implement reasonable technical requirements...". This is an unreasonable and overly broad restriction on the proposed remedy. The specific example given, failure of support ActiveX, is a most egregious example. ActiveX is not a feature of Windows, it is an API created for Internet Explorer middle-ware expressly to tie that middle-ware to the Windows platform. In a healthy competitive environment it should be end users that conclude if middle-ware is providing "functionality consistent with the Windows product", not Microsoft. The idea that Microsoft themselves are qualified to say what is and what is not a valid non-Microsoft middle-ware product puts the fox in charge of the henhouse. In fact by the definitions of this section of the Final Judgment, most existing successful non-Microsoft middle-ware (Java, Netscape Navigator, Web Plug-ins) would be exempt from the remedy. It was precisely the success of these products, demanded by end users, that precipitated the threat to Microsoft and led to the illegal behavior.

3.2 Limitations on disclosure of communications protocols

Section III.E. Requires disclosure of any communications protocol implemented in a Windows OS installed on a "client" computer. This would appear to exclude protocols implemented as Microsoft middle-ware, such as Web Browsers, or communications middle-ware such as e-mail programs (Outlook Express) or streaming media players (Windows Media Player). It would also appear to exclude protocols implemented in the same copy of Windows, running as a "server". Given the advent of "peer-to-peer" computing this distinction excludes more significant protocols than it includes. To meet the intent described in the impact statement, the requirement should be the disclosure of any communications protocol implemented by the Windows Operating System Product and any Microsoft middle-ware product.

3.3 Preventing disclosure on "security" grounds.

Section III.J.1.a attempts to limit the APIs and protocol descriptions to be published as part of the proposed remedy. The exceptions include those that would "compromise the security..." of the Microsoft products. It is well known and supported by the majority of reputable computer security experts, including many who work for Microsoft Corporation, that disclosure of the mechanisms of software makes it more secure, not less secure. In fact requiring Microsoft to document and disclose APIs will make the products more secure as flaws are discovered by peer review and then repaired. Computer security should not be considered valid technical grounds to limit disclosure.

3.4 Limitations on who can access the disclosures

Section III.J.2 places all kinds of limitations on the disclosure of the information central to the proposed remedy. In III.D the Final Judgment requires Microsoft to disclose APIs to all listed parties via "MSDN or similar" i.e. publicly and for a small fee. This

conflicts with III.J.2 which allows Microsoft to withhold such information unless Microsoft itself determines "a reasonable business need", or that the requester meets "standards established by Microsoft for ... viability". These restrictions are unnecessary and are not vital to the remedy. The required information should be disclosed simply, via MSDN or Microsoft.com, to anyone who has a valid Windows license.

Section III.J.2 additionally requires that non-Microsoft middle-ware innovators be in "compliance with Microsoft specifications" and, at their own expense, pass a Microsoft defined third party verification test. These new tests and requirements are onerous, and do not exist in the market today except as optional marketing programs. In particular the non-Microsoft middle-ware at issue in the anti-trust action would not have met these standards. These additional requirements and limitations will serve to place further hurdles in front of middle-ware ISVs. They only serve the interests of the monopolist in limiting access to the required APIs as has happened in the past as documented in the Findings of Fact.

4. Restrictive language related to Intellectual Property.

The licensing terms implied by the Final Judgment are both more onerous than the prevailing market today, and unfairly biased in favor of Microsoft.

The terms of III.G are not in force if Microsoft licenses intellectual property from the third party. This would appear to allow, for example, Microsoft to enter into an exclusive distribution arrangement with an ICP if the ICP had a reciprocal license to Microsoft for some middle-ware enhancement related to their Internet content. This kind of transaction is common in the industry today and would seem to weaken the intent of III.G

Section III.I.5 grants Microsoft the right to require a competitor to license to it IP rights to "relating to the exercise of their options or alternatives provided by this Final Judgment". This is an onerous and unreasonable requirement because Microsoft does not need such non reciprocal IP rights to comply with the Final Judgment. (Could such rights be licensed father by Microsoft to other ISVs?)

III.I requires Microsoft to reasonable and non discriminatory licensing of any intellectual property required for the market to take advantage of the provisions of the Final Judgment. However there is a restriction (H.III.3) on sub-licensing. This would in practice curtail most ISV business models if a technology innovator was unable to resell its technology to an "end user" OEM or ISV without that entity then being required to obtain a license from Microsoft.

The last paragraph of III.I explicitly states that the terms of the Final Judgment will not confer any rights with regard to Microsoft IP on anyone. But as the Final Judgment requires disclosure by Microsoft of APIs, protocols and detailed documentation of mechanisms inherent in middle-ware interfaces, then certain legal rights are in fact surrendered in most jurisdictions.

III.I does not address the significant and influential market in royalty free software (such as Linux) and the open standard nature of the Web protocols and standards. Industry standards groups which Microsoft itself is an active member of such as W3C (The World Wide Web Consortium) customarily require all APIs and protocols to be royalty free. Yet III.I potentially places further restrictions or costs on ISVs developing products and innovations under that model if they wish to integrate them with Windows.

5. Problems with the term and proposed implementation

5.1 Term is not long enough

The Final Judgment has a term of five years (V.A), or seven years with additional violations. Given the pattern of illegal behavior by Microsoft since 1995 and the fact that Windows Operating system product cycles are frequently many years apart, the scope of this agreement appears unusually short. A 10 or 15 year agreement would be more appropriate.

5.2 Issues with creating a competent technical body

The Final Judgment requires a three person technical committee. While this committee is intended to be knowledgeable about software design and programming, it also needs to be knowledgeable about Internet standards and protocols, online transactions and web e-commerce architectures and business models. It is unlikely that a committee as small as three people will have the requisite skill set to oversee the broad range of initiatives and innovations that center on the Windows platform and are the subject of the monopoly concern. The committee would be more in keeping with industry standards and accepted practice if it were larger and comprised of experts in several fields.

5.3 Public disclosure of information relating to enforcement

Section IV.B.10 and other language in IV (e.g IV.D.4.d) suggests that the Final Judgment requires the work of compliance and

technical overview to be conducted in secret. For example if an ISV submitted a complaint to the TC or the Microsoft Compliance Officer it is not required that the complaint and its response be published (IV.D.3) It would be more in keeping with industry standards and accepted practice for technical discussion around the enforcement of a Final Judgment be open to wider technical review. This would improve the quality and accuracy of such review as well as reassuring the community of OEMs, ISVs etc. that the enforcement process was actually working. At a minimum there should be a requirement that the TC host an independent web-site to communicate with the industry about the status of enforcement issues.

6. Flaws in several of the definitions

There are many problems with the definitions of key terms that affect the meaning and substance of the Final Judgment.

VI.A. A suitable definition for Application Programming Interface needs to include interfaces provided by middle-ware itself, since middle-ware can include tiers of software, not just a simple arrangement where middle-ware calls the Windows software layers. A more accurate and common definition of APIs would be independent of both the terms Windows and middle-ware.

VI.B. The scope of Communications Protocol should not be limited to communications with a "server operating system". This excludes the concept of one Windows XP PC talking to another PC, which is a common occurrence and should be within the scope of the remedy. "Peer-to-peer" is an example of a middle-ware category that is not covered by this definition.

VI.J.2 and VI.K.b.iii both require that the covered software be "Trademarked" to be under the terms of this agreement. This requirement seems to exclude certain middle-ware. For example "My Photos" and "Remote Desktop" are new middle-ware in WindowsXP and are apparently not trademarked. VI.T defines Trademarked to exclude certain named products regardless of their impact in the market.

VI.J.4 excludes software that has no user interface, such as a streaming video codec or a web commerce protocol handler.

VI.K.1 lists certain products explicitly as middle-ware. Given that the Final Judgment as written only covers Windows XP and subsequent versions (it should be modified to cover prior versions), the list of covered products and categories should also include MSN Explorer, Microsoft Outlook and other Microsoft Office components, Windows Movie Maker and others.

VI.N limits the definition of a "non-Microsoft middle-ware product" to one that has shipped 1,000,000 copies in a previous year. Under this definition, Netscape Communicator would not be covered by this Final Judgment, nor would Sun's Java JVM, both examples cited by the Court of middle-ware that require relief. The idea that a competing product has to already be successful to receive the protection of the Final Judgment is flawed. This condition should be removed.

VI.N defines non-Microsoft middle-ware in terms of code exposing APIs, which are defined in VI.A as being uses by Microsoft middle-ware (this is a circular definition). More importantly, non Microsoft middle-ware should not be defined more narrowly than Microsoft middle-ware. Not all middle-ware "exposes a range of functionality to ISVs though published APIs" although some (like Java) does. The original Netscape 1.0 web browser would have failed the definition in VI.N

VI.Q defines Personal Computer as using an Intel x86 processor. Microsoft has in the past and will most likely in the future ship Windows Operating systems for processors other than x86. The Court found that Microsoft's illegal practices in respect of distribution of Internet Explorer also extended to the Macintosh Power-PC platform so this definition is overly narrow.

VI.R. 150,000 beta testers is an unusually large number, even for Windows and suggests that "timely manner" would be defined as the last test release of a Microsoft product rather than the first public test release. The interests of the enforcement are better served if Timely Manner was defined as the first public test release of a Windows OS product.

[end]